

CG における流体と弾性体の連成シミュレーションに関する研究

Animating the Interaction between Deformable Bodies and Fluid

指導教員 河口 洋一郎 教授

60816 今村 紀之

1. 序論

近年、映画や CM、ゲームなどをはじめ、コンピュータ・グラフィックス(CG)とその応用分野において、物理シミュレーションの需要が高くなってきている。これは物理法則に裏付けされた物体の挙動を手でモデリングするのが困難なためである。特に流体や弾性体などの連続体はモデリングが難しく、シミュレーション手法に対する需要は極めて高い。

流体や弾性体をそれぞれ計算する基礎的なシミュレーション手法は数多く開発されているが、両者のインタラクションを扱った高速な手法は未だ提案されていない。

以上の背景を受け、本研究では、流体と弾性体の高速かつ安定な物理ベースの連成シミュレーション手法の提案を目的とする。

2. 流体領域のモデル化

流体領域は、スタガード・グリッドを用いて離散化し、移流項の計算に Semi-Lagrange 法を用いる¹⁾。

2.1 支配方程式

流体の支配方程式として非圧縮性ニュートン流体における Navier-Stokes 方程式(2.1)と連続の式(2.2)を用いる。

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla) \mathbf{u} - \frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{f} \quad (2.1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (2.2)$$

2.2 流体シミュレーションの特徴

本研究の流体領域では、高速に計算可能な一様正方グリッドを用いて空間の離散化を行う。

また、移流項には Semi-Lagrange 法を用い、粘性拡散項および圧力ポアソン方程式は陰的に

計算する。連立方程式の解法には Jacobi 反復法を用いる。さらに、圧力振動を抑えるため、ベクトルの物理量をグリッドの面上に定義するスタガード・グリッドを用いる。

2.3 射影

Helmholtz 射影 $\mathcal{P}$ を用いることにより、式(2.1)の移流項等を計算した後の divergence-free ではない速度場を divergence-free な速度場にする。

2.4 計算の手順

各タイムステップでの流体の計算は以下のようになる。

- (1) 移流項・粘性拡散項・圧力項の計算
- (2) (1)までの部分速度 $\mathbf{w}$ を用いた圧力ポアソン方程式(2.3)の計算
- (3) (2)で得た圧力 p を用いた射影(2.4)の計算

$$\nabla^2 p = \nabla \cdot \mathbf{w} \quad (2.3)$$

$$\mathbf{u} = \mathbf{w} - \nabla p \quad (2.4)$$

3. 弾性体のモデル化

弾性体領域には、explicit FEM²⁾ を用い、応力の計算および時間積分をどちらも陽的に行う。

3.1 ひずみとひずみ速度

弾性体のひずみの定義として Green のひずみ(3.1)を用いる。

$$\epsilon_{ij} = \frac{\partial \mathbf{x}}{\partial m_i} \cdot \frac{\partial \mathbf{x}}{\partial m_j} - \delta_{ij} \quad (3.1)$$

ここで、変位後のワールド座標での位置 $\mathbf{x}$ は、物質座標で表された物体の初期座標 $\mathbf{m}$ の関数としてあらわされるものとする。



図 1 上端固定条件での弾性体と噴出流体の連成シミュレーション[速度場による可視化, 左下は実時間]

また、ひずみ速度 $\dot{\epsilon}$ については(3.1)の両辺を時間微分することにより得られる。

3.2 構成式

弾性体のモデルとして、バネとダッシュポットが並列に並んだモデルを考える。このとき、応力 σ は、ひずみに影響する項 $\sigma^{(\epsilon)}$ とひずみ速度に影響する項 $\sigma^{(\dot{\epsilon})}$ の和であるとみなすことができる。両者はそれぞれ弾性および減衰ポテンシャルを空間微分することで得られ、定数 λ 、 μ 、 λ' 、 μ' を用いてそれぞれ、式(3.4)、式(3.5)のようにならわされる。

$$\sigma_{ij}^{(\epsilon)} = \lambda \text{tr}(\epsilon) \delta_{ij} + 2\mu \epsilon_{ij} \quad (3.4)$$

$$\sigma_{ij}^{(\dot{\epsilon})} = \lambda' \text{tr}(\dot{\epsilon}) \delta_{ij} + 2\mu' \dot{\epsilon}_{ij} \quad (3.5)$$

3.3 空間の離散化

弾性体領域は2次元の計算では、三角形メッシュによる空間離散化を行う。各三角形要素での物理量は重心座標系であらわされる。

3.4 計算の手順

各タイムステップでの計算手順を以下に示す。

- (1) 変形勾配・変形勾配速度テンソルの計算
- (2) 弾性応力 $\sigma^{(\epsilon)}$ ・粘性応力 $\sigma^{(\dot{\epsilon})}$ の計算
- (3) 内力・外力の計算
- (4) 速度・変位の計算

なお、境界部で流体から受けた力は、手順(3)で外力として足し合わせる。

4. 境界領域のモデル化

4.1 概要

通常、剛体と流体等での高速な連成手法では、境界を流体グリッドに合わせて階段状に近似することが多い³⁾。しかし、これを弾性体に適用しようとする、流体のすべての計算点で弾性体の各要素との内外判定を行わなければならない、計算コスト的に実用に耐えない。

本手法では、境界マップという概念を導入し、流体計算点から境界までの位置のみを差分スキームに取り込むことで、従来手法より高速で精度の高い連成シミュレーションを実現する。

4.2 境界マップ

境界マップとは、流体グリッドと弾性体の境界辺との交点に、弾性体の境界辺とその速度の情報を持たせ、流体グリッド上に関連付けたものである。

境界マップは境界辺すべてについて、流体グリッドとの交点を求めることで実現されるため、計算コストが低い。

4.3 内部判定

ソートされた境界マップを順に参照し、奇数番目の要素と偶数番目の要素の間を内部としてビットを立てることで、内部判定を行う。

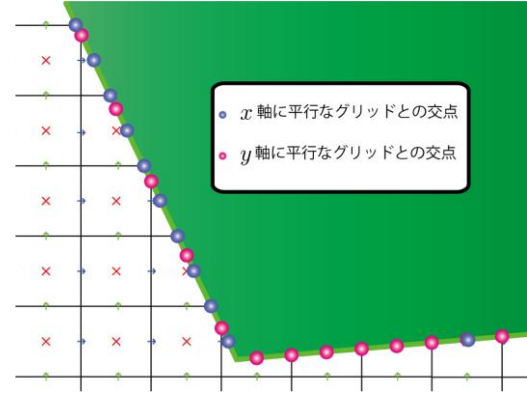


図 2 x軸、y軸方向の境界マップの作成

4.4 境界条件の適用

流体の計算は物体の外部にある計算点のみで計算を行う。この際、内部判定の結果をもとに、計算に用いる隣接計算点が物体の内部にあるか外部にあるかが判断できる。

ここで、境界上では流体の速度が物体の速度と同じであるという条件を用いると、隣接計算点が利用できない場合にもこの条件との連立により式を導くことができる。

4.5 境界に作用する力の計算

境界部に作用する力は、境界条件を自然に満たすように定式化を行う。各流体計算点について、その計算点で算出された速度を u_P 、近傍の境界マップに格納された点 Q での速度を $\hat{u}$ とすると Q に働く力 f は以下のようにモデル化する。

$$f = \frac{u_P - \hat{u}}{2\Delta t} \quad (3.4)$$

5. 結果

本手法を用いて、矩形状の棒の上端を固定し、下端を流体の噴出にさらすシミュレーション実験を行った。速度場と弾性体の挙動を可視化した結果を図1に示す。

流体の噴出により棒が変形し、流体の速度場に影響を与える様子が見て取れる。なお、シミュレーションに要した時間は、レンダリングも含め83ms程度である。

6. 結論

境界マップの概念を導入することにより、流体と弾性体の高速な連成シミュレーションを実現した。

参考文献

- 1) J. Stam, In *Siggraph Proc.* (1999).
- 2) J. F. O'Brien et al., In *Siggraph Proc.* (1999).
- 3) K. Crane et al., In *GPU Gems 3*, [30], (2007).